**第十三章**

继承通过使用已有的类（基类）定义新的类（派生类）使得能够根据需要修改编代码。公有继承建立is-a关系，这意味着派生类对象也应该是某种基类对象。作为is-a模型的一部分，派生类继承基类的数据成员和大部分方法，但不继承基类的构造函数、析构函数和赋值运算符。派生类可以直接访问基类的公有成员和保护成员，并能够通过基类的公有方法和保护方法访问基类的私有成员。可以在派生类中新增数据成员和方法，还可以将派生类用作基类，来做进一步的开发。每个派生类都必须有自己的构造函数。程序创建派生类对象时，将首先调用基类的构造函数，然后调用派生类的构造函数;程序删除对象时，将首先调用派生类的析构函数，然后调用基类的析构函数。

如果要将类用作基类，则可以将成员声明为保护的，而不是私有的，这样，派生类将可以直接访问这些成员。然而，使用私有成员通常可以减少出现编程问题的可能性。如果希望派生类可以重新定义基类的方法，则可以使用关键字vitual将它声明为虚的。这样对于通过指针或引用访问的对象，能够根据对象类型来处理，而不是根据引用或指针的类型来处理。具体地说，基类的析构函数通常应当是虚的。可以考虑定义一个 ABC：只定义接口，而不涉及实现。例如，可以定义抽象类Shape，然后使用它派生出具体的形状类，如Circe和Square。ABC必须至少包含一个纯虚方法，可以在声明中的分号前面加上=0来声明纯虚方法。

virtual double area( ) const = 0;

不一定非得定义纯虚方法。对于包含纯虚成员的类，不能使用它来创建对象。纯虚方法用于定义派生类的通用接口。